

**In the Claims:**

The claims as previously amended are presented below for reference:

1. (Previously Presented) A method for evaluating compliance with a service-level agreement, said method comprising the steps of:

defining classes of service for at least one of host system, user, URL, hosted site, transaction, content and file type;

defining a set of parameters to be measured for each of said classes;

defining acceptance levels for each parameter in said set of parameters;

collecting real-time information related to measurement of said parameters; and comparing said acceptance levels to said real-time information.

2. (Previously Presented) The method of claim 1 in which said step of defining further includes the steps of:

providing a format in which a set of servers will provide information to be measured; and implementing a means for collecting said information.

3. (Original) The method of claim 1 in which said step of defining acceptance levels comprises the step of generating a database entry for each service commitment element of a service-level agreement.

4. (Previously Presented) The method of claim 1 in which the set of parameters to be measured is selected from the set consisting of: records of performance, and errors, client IP address, username, date, time, service, server name, server IP address, processing time, bytes sent, bytes received, service status, operation, target URL, User Agent, referrer parameters, SMserver, Smvirtual site, and cookie.

5. (Previously Presented) The method of claim 1 wherein the real-time information collected further includes information selected from the group consisting of: assigned disk space,

what the user can access, how the user's request is fulfilled within the system or web farm, user's subscribed level of service or class, transaction, number of requests, download size, file size, file type, time of day, week or month, response time of the back end servers, response time of the web farm, and how long it takes to complete a specified request or file.

6. (Currently Amended) An apparatus for evaluating real-time compliance with a service-level agreement comprising:

at least one back-end server ~~and a reporter~~;

a network connecting said at least one back-end server ~~and said reporter~~;

a collection processor measuring and periodically collecting a set of defined parameters for said at least one back-end server;

a set of acceptance levels for said collected parameters wherein said acceptance levels depend on ~~at least one of~~ a class of service for at least one of host system, URL, hosted site, transaction, content, file type and user;

a monitoring processor determining which of said collected parameters exceed a corresponding acceptance level; and

a reporting process that produces a report the results of said monitoring processor.

7. (Previously Presented) The apparatus of claim 6 in which the collection processor further comprises:

an intelligent agent deployed on each of said back-end servers monitoring a set of defined parameters and logging them into respective log files;

a scheduler triggering said reporter to begin collection of log files from a list of back-end servers;

an accumulator requesting log files from the intelligent agent of each listed back-end server and consolidating the log files into a database; and

an interface mechanism between said accumulator and each intelligent agent, said interface mechanism ensuring that each requested log file is completely transferred to the accumulator prior to starting consolidation.

8. (Previously Presented) The apparatus of claim 7 wherein each of said intelligent agents further keeps track of which portions of said log files have been transferred.

9. (Previously Presented) A method for implementing management of a service level agreement monitoring system for a set of back-end servers in a web farm comprising the steps of:

defining classes of service parameters according to a hierarchy of service levels;  
selecting at least one class of service parameters to be monitored;  
creating a database of monitored service parameters; and  
preparing reports and/or alarms according to said selected at least one class of service parameters.

10. (Previously Presented) The method of claim 9 further including the step of providing a graphical user interface for performing at least one of selecting the class of service parameters to be monitored, defining thresholds of service-level commitments for at least some of said service parameters, defining alarm trigger events, scheduling monitoring and reporting functions, and determining reporting formats.

11. (Previously Presented) The method of claim 10 wherein said step of preparing reports and/or alarms is further based on thresholds, schedules, and formats defined by the graphical user interface.

12. (Previously Presented) The method of claim 9 in which said service parameters to be monitored are selected from the set consisting of: records of performance, transactions, and errors, client IP address, username, date, time, service, server name, server IP address, processing time, bytes sent, bytes received, service status, operation, target URL, User Agent, referrer parameters, SMserver, Smvirtual site, and cookie.

13. (Currently Amended) The method of claim 9 wherein said database of monitored parameters further includes information selected from the group consisting of: the assigned disk space, what the user can access, how the user's request is fulfilled within the system or web farm, user's subscribed level of service or class, transaction, number of requests, download size, file size, file type, time of day, week or month, response time of the back end servers, response time of the web farm, and how long it takes to complete a specified request or file.

14. (Previously Presented) The method of claim 9 in which said classes of service parameters are selected according to user class, host class, and virtual site class.

15. (Previously Presented) The method of claim 9 wherein the step of defining classes is based on one of users, URLs and virtual sites.

16. (Previously Presented) The method of claim 10 wherein the step of defining classes is based on one of users, URLs and virtual sites.